Claims: I claim

1. A method for transmitting the time-critical data over Ethernet networks; the method comprising:

   A scheme of carrying the redundant data in each packet, which can be used to recover the lost information when a packet is dropped;

   A strategy of actively and selectively dropping packets when congestion happens to resolve the traffic congestion, the lost information will be able to recovered from the redundant data; and

   A means of recovering the lost information when packet-drop happens, so that the quality of transmission will not be affected.

2. The method of claim 1, wherein said the scheme of carrying the redundant data in each packet can be done as follows. The data carried in each packet is composed of two parts: the required data and the redundant data. The required data is the data that the packet supposes to carry in the case without redundant data. The redundant data is the required data of the next packet. So that if the next packet is dropped we can recover the lost information from the redundant data of this packet.
3. The method of claim 1, wherein said the strategy of actively and selectively dropping packets when congestion happens can be done as follows. When congestion happens, we drop all the even-number packets. All the lost data can be recovered from all the odd-number packets' redundant data.
4. The method of claim 1, wherein said the strategy of actively and selectively dropping packets when congestion happens can be done as follows. When congestion happens, we drop all the odd-number packets. All the lost data can be recovered from all the even-number packets' redundant data.
5. The method of claim 1, wherein said the means of recovering the lost data when packet-drop happens can be accomplished as follows. When a packet arrives, we use the required data and save the redundant data. If the next packet is dropped, we use the saved redundant data as if the next packet was received. If the next packet arrives, we discard the saved redundant data.
6. A method for transmitting the time-critical data over packet-switched networks; the method comprising:

A scheme of carrying the redundant data in each packet, which can be used to recover the lost information when a packet is dropped;

A strategy of actively and selectively dropping packets when congestion happens to resolve the traffic congestion, the lost information will be able to recovered from the redundant data; and

A means of recovering the lost data when packet-drop happens, so that the quality of transmission will not be affected.

7. The method of claim 6, wherein said the scheme of carrying the redundant data in each packet can be done as follows. The data carried in each packet is composed of two parts: the required data and the redundant data. The required data is the data that the packet supposes to carry in the case without redundant data. The redundant data is the required data of the next *N* packets, where $N = 1, 2, 3....$ So that if any of the next *N* packets are dropped we can recover the lost data from the redundant data of the received packet.
8. The method of claim 6, wherein said the strategy of actively and selectively dropping packets when congestion happens can be done as follows. When congestion happens, we keep one packet and drop *N* packets that follow the kept packet, where $N = 1, 2, 3....$ All the lost data can be recovered from the redundant data of the kept packet.
9. The method of claim 6, wherein said the means of recovering the lost data when packet-drop happens can be accomplished as follows. When a packet arrives, we use the required data and save the redundant data. If the next *i* packets are dropped, we will use the first *i* parts of the redundant data of the received packet to recover the lost data, where $i = 1, 2, ... N$. If no packet is dropped, we only use the required data and discard the saved redundant data.

A Packet-drop Tolerant Method for Transmitting Time-critical Data over Ethernet

Abstract: A method for transmitting the time-critical data over Ethernet. The method can tolerate at least 50% of packet-drop without affecting the quality of transmission. Ethernet network is considered a connectionless network, so the packet-drop is not avoidable. Therefore, the key to transmitting the time-critical data over Ethernet is not to prevent the packet-drop, but to control and to tolerate the packet-drop. To tolerate the packet-drop, we have to carry redundant data in each Ethernet packet so that we can recover the lost information from the redundant data when packet-drop occurs. An easy way of carrying the redundant data is as follows. Every packet carries two parts of data. The first part is the required data, and the second part is the redundant data. The required data is the data that the packet is supposed to carry in the case without redundant data. The redundant data is the required data of the next packet. When a packet is dropped, we can use the redundant data of the previous packet to recover the lost information. To be able to recover all the lost information, we need to control the packet-drop. In other words, when congestion happens, we should actively and selectively drop the packets to prevent random and bursty packet-drop so that the redundant data can be used to recover all the lost information. Based on the redundant data we are carrying, one way of controlling the packet-drop is as follows. When congestion happens, we drop all the even-number packets, and all the lost information can be recovered from all the odd-number packets' redundant data. This is a 50% of traffic reduction, and we don't affect the quality of transmission at all.

It looks like we doubled the Ethernet traffic by carrying the redundant data. But our calculations show that, in our example, for a 50% of packet-drop tolerance, we only increase 10% of the Ethernet traffic when we use RTP. If we use cRTP, we won't increase the Ethernet traffic at all; this means we get 50% of packet-drop tolerance for free when using cRTP.

The method doesn't require any physical or wiring changes to the existing Ethernet. The transmission of the regular best-effort data will be the same as the standard Ethernet. Finally, it's a software only solution and easy to implement.

References

[1] Charles E. Spurgeon "Ethernet, The Definitive Guide" 2000, O'Reilly & Associates, Inc., 101 Morris Street, Sebastopol, CA 95472 USA

[2] Jonathan Davidson, James Peters "Voice over IP Fundamentals" 2000, Cisco Press, 201 West 103rd Street, Indianapolis, IN 46290 USA